

We claim:

1. A method of executing a plurality of threads within a single programmable processor, the method comprising:

receiving an instruction stream for each one of the plurality of threads at an execution

5 unit;

executing instructions from each instruction stream received at the execution unit in a multistage pipeline within the execution unit such that, at any given time, the multistage pipeline includes instructions from different ones of the instruction streams in different stages of the multistage pipeline, the instructions including a single instruction that operates on a plurality of  
10 data elements in partitioned fields of at least one register to produce a catenated result, the at least one register having a register width and each of the data elements having an elemental width smaller than the register width.

2. The method of claim 1 wherein the number of threads executing within the execution unit

15 is prime relative to a rate of execution of a slowest functional unit in the execution unit.

3. The method of claim 1 wherein the instructions from the plurality of instruction streams are executed in a round-robin manner.

20 4. The method of claim 1 wherein only one thread from the plurality of threads can handle an exception at any given time.

5. The method of claim 1 further comprising:

decoding a second single instruction specifying a third and a fourth register each

containing a plurality of floating-point operands;

    multiplying the plurality of floating point operands in the third register by the plurality of operands in the fourth register to produce a plurality of products; and

    providing the plurality of products to partitioned fields of a result register as a catenated

5     result.

6.     A computer-readable medium:

    having an instruction stream for each one of a plurality of threads that instruct a computer system to perform operations comprising,

10     receiving an instruction stream for each one of the plurality of threads at an execution unit;

    executing instructions from each instruction stream received at the execution unit in a multistage pipeline within the execution unit such that, at any given time, the multistage pipeline includes instructions from different ones of the instruction streams in different stages of the multistage pipeline, the instructions including a single instruction that operates on a plurality of data elements in partitioned fields of at least one register to produce a catenated result, the at least one register having a register width and each of the data elements having an elemental width smaller than the register width.

20     7.     The computer-readable medium of claim 6 wherein the number of threads executing within the execution unit is prime relative to a rate of execution of a slowest functional unit in the execution unit.

8. The computer-readable medium of claim 6 wherein the instructions from the plurality of instruction streams are executed in a round-robin manner.

9. The computer-readable medium of claim 6 wherein only one thread from the plurality of threads can handle an exception at any given time.

10. A computer data signal, embodied in a transmission medium:

having an instruction stream for each one of a plurality of threads that instruct a computer system to perform operations comprising,

receiving an instruction stream for each one of the plurality of threads at an execution unit;

executing instructions from each instruction stream received at the execution unit in a multistage pipeline within the execution unit such that, at any given time, the multistage pipeline includes instructions from different ones of the instruction streams in different stages of the multistage pipeline, the instructions including a single instruction that operates on a plurality of data elements in partitioned fields of at least one register to produce a catenated result, wherein the at least one register has a register width and each of the data elements has an elemental width smaller than the register width.

11. The computer data signal of claim 10 wherein the number of threads executing within the execution unit is prime relative to a rate of execution of a slowest functional unit in the execution unit.

12. The computer data signal of claim 10 wherein the instructions from the plurality of instruction streams are executed in a round-robin manner.

13. The computer data signal of claim 10 wherein only one thread from the plurality of threads can handle an exception at any given time.

14. The computer data signal of claim 10 wherein at least some of the instructions further include a group floating point multiply instruction for multiplying floating point data in a programmable processor; the group floating point multiply instruction capable of instructing the computer to perform operations comprising:

decoding the group floating point multiply instruction specifying a third and a fourth register each containing a plurality of floating-point operands;

multiplying the plurality of floating point operands in the third register by the plurality of operands in the fourth register to produce a plurality or products; and

providing the plurality of products to partitioned fields of a result register as a catenated result.